



SITE D'AVIS DE RESTAURANTS

SOMMAIRE

- Cahier des charges
- Conception graphique
- Framework Vue.js
- Les API
- Architecture de l'application
- Le Store



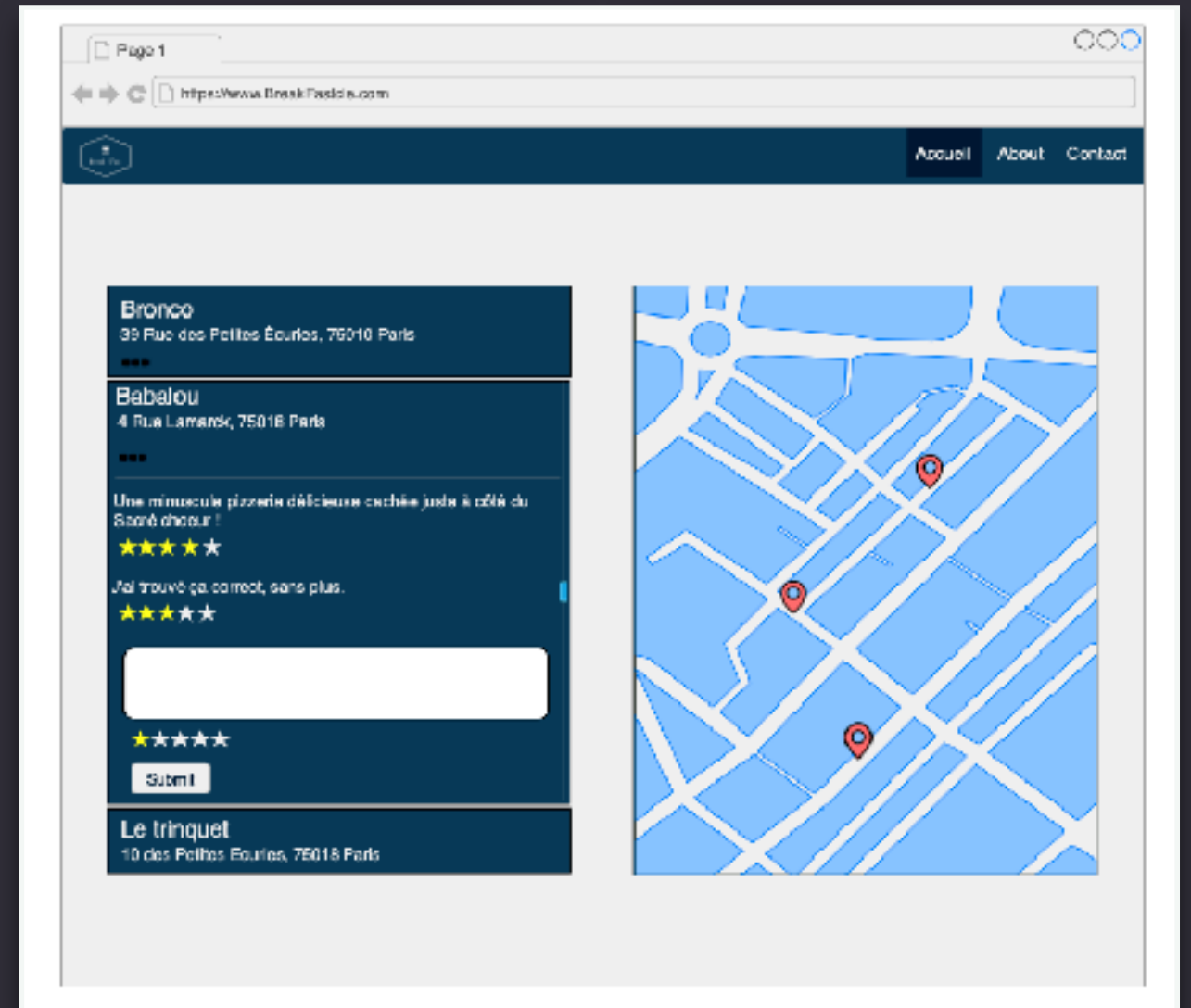
CAHIER DES CHARGES

Carte de restaurants

Une carte Google (API Google Map) doit être centrée sur la position de l'utilisateur (API Geolocation Javascript) et lister les restaurants présents dans la zone sélectionnée (marqueurs, API Google Places) grâce à leurs coordonnées. La liste des restaurants doit être visible aux côtés de la carte et renseigner les informations disponibles, une photo (API Google Street View Static) ainsi que les commentaires et notes. Un filtre doit être disponible pour trier les restaurants suivant leur moyenne (entre 1 et 5).

Ajouter des restaurants et des avis

L'utilisateur doit pouvoir ajouter un avis sur un restaurant déjà existant ainsi qu'un nouveau restaurant (position marqueur). Toutes les modifications apportées par l'utilisateur doivent disparaître à la fermeture ou rafraîchissement de la page.



Maquette de départ

CONCEPTION GRAPHIQUE

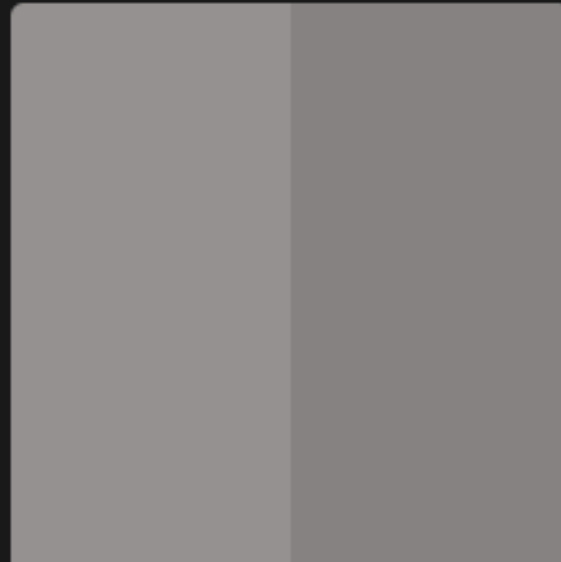
Poppins

AMATIC SC



WHITE LILAC

#EAEAED



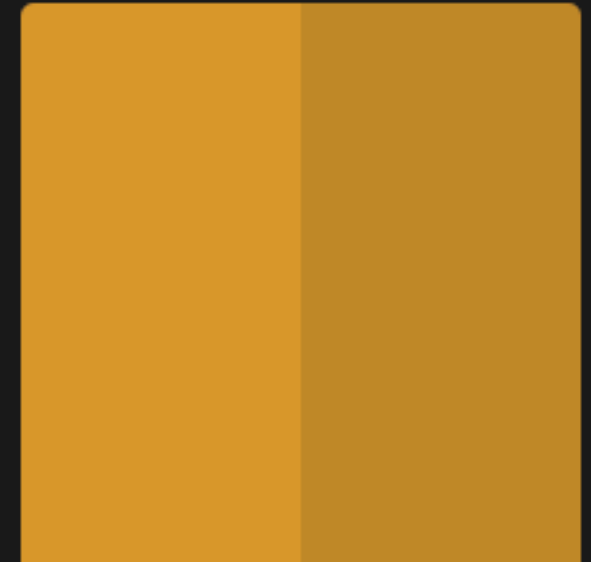
MOUNTAIN MIST

#969492



GREEN WATERLOO

#23241c



BUTTERCUP

#da972e

#EAEAED

WHITE LILAC

#969492

MOUNTAIN MIST

#23241c

GREEN WATERLOO

#da972e

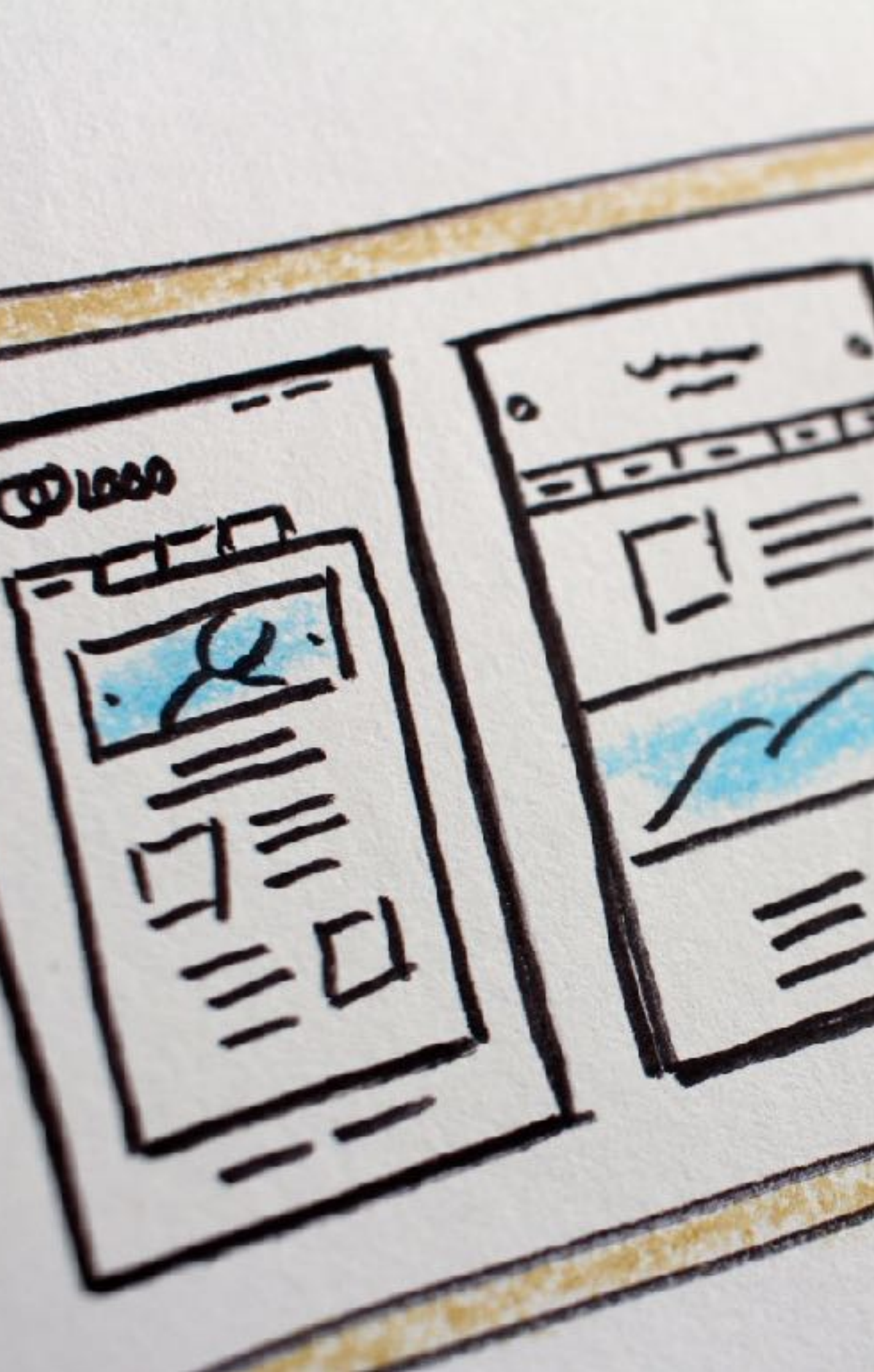
BUTTERCUP

VUE.JS

Vue.js est un framework JavaScript open-source utilisé pour construire des interfaces utilisateur.

Vue.js propose son propre module router, **VueRouter**, qui va créer des « chemins » pour relier ces composants entre eux et fluidifier la navigation.

Vuex est un model d'implémentation, il permet de garder à jour l'état des données de l'application. Il centralise le stockage des données, facilite le rendu de celles-ci et leurs modifications.



MAP ET API

Les **API** sont un moyen de communication entre deux logiciels. Elles fonctionnent sur le principe de la communication entre client et serveur: le client formule une requête pour une obtenir une information et le serveur envoie une réponse contenant les données.

Comme précisé dans le cahier des charges, les API suivantes ont été intégrées à l'application:

- Geolocation API JavaScript: récupère la position de l'utilisateur.
- Google Map: génère une carte google interactive.
- Google Places (requêtes incluses: NearbySearch & Details): Permet d'identifier des lieux sur la carte autour d'un périmètre défini et d'obtenir des infos et renseignements.
- Street View Static: permet d'obtenir une image statique de la vue depuis le lieu sélectionné.



LE STORE

Le **Store** est composé d'un **State** qui sert de "source de vérité".

Les **getters** vont calculer des valeurs basées sur le state du store.

Les **mutations** sont les seules à pouvoir modifier l'état du store.

Les **actions** sont similaires aux mutations à la différence qu'elles ne peuvent modifier le state (seulement acter les mutations).

ARCHITECTURE D'APPLICATION

